# WHEN TO UNPAIR: REGULATING PAIRING DEPENDENCE IN MEDICAL VISUAL IN-CONTEXT LEARNING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Visual in-context learning (ICL), well suited to label-scarce medical imaging, uses support image–label pairs to demonstrate input–output mappings, while the labels collectively indicate the requested task. We diagnose dependence on individual pairings with a test-time *derangement* that reassigns every support label to another support image while preserving the query, support images, and label multiset. The resulting *pairing gap*, defined as shuffled-minus-matched performance, shows that all four released models depend on the pairing, to widely varying degrees. Further analysis of a paired-trained model reveals support-associated spurious regions and lesion-size biases even with real, unaltered supports, alongside sensitivity to mis-registered support labels. To regulate this dependence, we introduce a *late unpairing curriculum* (LUC), which starts with matched training and then applies *random unpairing*, replacing each support label with that of another support in the same episode. LUC nearly closes the pairing gap on two backbones while maintaining or improving matched-support performance across all evaluated task types, with gains extending to held-out tasks and cross-dataset episodes. It also mitigates these failure modes. On BraTS whole-tumor segmentation, matched-support DSC rises from $0.733$ to $0.857$ while the gap shrinks from $-0.184$ to $-0.008$. In a released model, brief fine-tuning with random unpairing reduces the gap. A reversed curriculum that places the same number of unpairing epochs at the start of training leaves a large gap. This shows that pairing dependence is shaped by the order of training and not only by the amount of unpaired training.

## 1 INTRODUCTION

In-context learning (ICL) lets a single model solve a new task from a few examples, with no task-specific retraining (Brown et al., 2020). In its visual form the model receives a query image and a few support (image, label) pairs and returns a dense prediction, such as a segmentation or an image in another modality (Bar et al., 2022; Wang et al., 2023a;b; Kim et al., 2023). This suits medical imaging, where dense labels are scarce and expensive, and medical ICL models now cover 2D and 3D segmentation and image-to-image tasks (Butoi et al., 2023; Czolbe & Dalca, 2023; Rakic et al., 2024; Hu et al., 2025; 2026).

An underexplored question in ICL is how models use the supports to infer the task. Support pairs provide task-relevant information in complementary ways. A model can use the image–label correspondence within individual support pairs to infer how inputs map to outputs, while the collection of support labels can indicate what output is requested. For example, support masks can collectively distinguish whole-tumor from edema segmentation, while their pairing with individual images provides example-specific mappings between image appearance and label. Both sources can be informative, but how current visual ICL models balance them, and how strongly their predictions depend on individual image–label pairings, remains poorly understood.

We study this behavior through a test-time derangement that selectively disrupts individual image–label correspondence. Specifically, we preserve the query image, all support images, and the same set of support labels, but reassign each label to a different support image, which gives shuffled supports (Figure 1). Comparing performance under matched and shuffled supports yields a pairing gap, which quantifies a model's dependence on individual support pairings while holding the contents of the support set fixed. In doing so, we find substantial variation in pairing dependence across



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

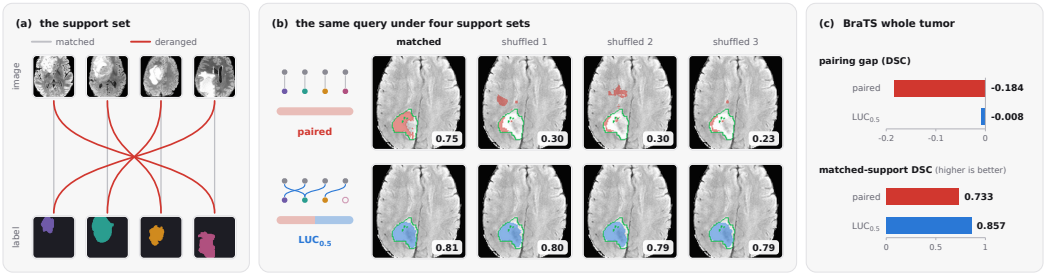


Figure 1: **Which label sits on which support image changes the prediction.** (a) Grey: every support image keeps its own label; red: a *derangement*, which moves each label onto another image and leaves the label multiset unchanged. (b) One BraTS validation case at $K{=}4$ with matched supports and three derangements; beside each row are the support set that model trains on and its schedule $p(e)$. Green outlines the ground truth; numbers are DSC. (c) Pairing gap and matched-support DSC over the BraTS whole-tumor evaluation.

existing visual ICL models (Table 1) and show that strong dependence can have undesirable consequences. In medical image segmentation, for example, the predicted lesion extent and location can be influenced by the size and spatial position of lesions in the support labels (Section 4).

Building on this diagnostic, we find that training-time random unpairing can regulate how models use support pairing. Replacing support labels during training with those of other supports in the same episode weakens the reliability of individual image–label correspondence, while the labels, taken together, still indicate which output is requested. The timing of this intervention matters: a late unpairing curriculum (LUC), which unpairs only after an initial period of conventional paired training, substantially reduces test-time pairing dependence. This robustness comes without a trade-off in standard predictive performance: LUC maintains or improves performance under correctly matched supports across all evaluated task types. It also mitigates several failure modes associated with excessive dependence on individual support pairs. Together, the test-time derangement and training-time unpairing form a strategy for diagnosing and controlling pairing dependence in visual ICL: the former reveals how a trained model uses support correspondence, and the latter reshapes that behavior without changing the model architecture or giving up matched-support performance.

Our contributions are:

- A test-time derangement of the support labels as a black-box diagnostic of pairing dependence, and the finding that this dependence varies widely across four released visual in-context learners (Section 4).
- The failure modes of strong dependence in paired training: the prediction follows the support labels rather than the query, with real support sets and under controlled perturbations of the labels, and loses accuracy when the support labels are mis-registered (Section 4).
- LUC, a schedule for random unpairing that substantially reduces pairing dependence while maintaining or improving matched-support performance on five task types and two backbones, and that also reduces these failure modes (Section 5). The timing matters: with the same number of unpairing epochs, the schedule that ends on matched supports keeps a large gap. On a released model, 1000 fine-tuning steps of random unpairing shrink the edema gap from $-0.353$ to $-0.036$.

## 2 Related Work

**In-context learning.** In-context learning lets a model solve a task from a few input–output examples placed in its input (Brown et al., 2020). In language models, how much the example labels matter depends on the setting: random labels barely lower accuracy on classification tasks (Min et al., 2022), while larger models follow flipped labels (Wei et al., 2023) and ICL does learn the label relationships in its context (Kossen et al., 2024). Recognizing which task the examples name can be separated from learning the task in context (Pan et al., 2023), transformers can solve ICL




Figure 2: **The two operations on the support set.** (a) Matched supports. (b) The test-time derangement: the same four labels, none on its own image. (c) Random unpairing in training: every support takes a copy of another support's label, so a label can repeat, which never happens in (b).

by copying from matching context (Olsson et al., 2022), and the training data decide whether ICL emerges and whether it persists (Chan et al., 2022; Singh et al., 2023). We study pairing dependence in visual ICL and how it changes when support labels are reassigned during training.

**Visual and medical in-context learning.** Visual ICL defines a dense prediction task through a few (image, label) examples, first by inpainting a canvas that holds the examples and the query (Bar et al., 2022; Wang et al., 2023a) and then at scale and for segmentation and other dense tasks (Bai et al., 2024; Wang et al., 2023b; Kim et al., 2023). Segment Anything (Kirillov et al., 2023) can also be adapted to segment a target from one reference image and its mask (Zhang et al., 2024; Liu et al., 2024), and which examples are shown changes the result (Zhang et al., 2023). In medical imaging, where labels are scarce, few-shot segmentation (Ouyang et al., 2020) has grown into ICL models that solve unseen tasks from a support set without retraining: UniverSeg and Tyche in 2D (Butoi et al., 2023; Rakic et al., 2024), and Neuralizer, Neuroverse3D, Medverse and Iris for multi-task and 3D imaging (Czolbe & Dalca, 2023; Hu et al., 2025; 2026; Gao et al., 2025). Universal models that take a spatial or text prompt instead of examples form a parallel line (Ma et al., 2024; Wang et al., 2025; Liu et al., 2023). Support labels are drawn by hand and vary within and between observers (Karimi et al., 2020), and in-context segmenters lose accuracy when their support labels are deformed (Sheng et al., 2025). The ICL models are trained and evaluated with matched supports; we measure what four released models do when the pairing changes, and we change the pairing during training.

**Training on support sets.** Episodic training builds each step from a support set and a query (Vinyals et al., 2016; Snell et al., 2017). When a query can be solved without its supports, a meta-learner can learn to ignore them, which a random assignment of labels per task prevents (Yin et al., 2020; Rajendran et al., 2020). Random unpairing acts inside one support set and targets the opposite failure, excessive reliance on individual pairings: while it is applied, the labels still indicate the task as a set, but which image each label sits on carries no information.

**Curriculum learning.** Curriculum learning orders the training signal, classically from easy to hard examples (Bengio et al., 2009), and later work chooses the order automatically and studies when it helps (Graves et al., 2017; Hacohen & Weinshall, 2019; Wu et al., 2021; Soviany et al., 2022). The order also shapes what a network keeps: deficits early in training can have lasting effects (Achille et al., 2018). LUC orders how reliable the support pairs are rather than how hard the examples are, starting on matched supports and ending on unpaired ones; for the pairing gap we find the reverse of a lasting early effect, since the phase that ends training sets the gap.

## 3 METHOD

We measure a model's dependence on support pairings with a test-time derangement and then propose training-time random unpairing to regulate this dependence. Figure 2 shows both operations on a support set of four, and Section A gives the full protocol. We describe the method in the context of Neuroverse3D (Hu et al., 2025), but the same formulation applies to all models tested here.

**The visual in-context forward pass.** Given a query $x_q$ and a support set $S = \{(x_i, y_i)\}_{i=1}^{K}$, Neuroverse3D uses a shared-weight support branch $g$ to process each pair $(x_i, y_i)$ and predicts a dense label $\hat{y}_q$ with a target encoder $f_{\text{enc}}$ for the query. Support representations are combined by an

equal-weight mean at every decoder stage $s$,

$$\bar{c}^{\,s} \;=\; \frac{1}{K}\sum_{i=1}^{K} g_s\big(x_i \oplus y_i,\; f^{s}_{\text{enc}}(x_q)\big), \tag{1}$$

where $\oplus$ is channel concatenation, and a target decoder fuses $\bar{c}^{\,s}$ with skip connections to produce $\hat{y}_q$ ($\sim$71M parameters). The mean in Equation (1) is invariant to reordering whole support pairs but not to permuting their labels, since each image and its assigned label interact inside $g_s$ before aggregation. Any dependence on the matching therefore comes from the training distribution rather than from the aggregation.

**Test-time derangement of the support labels.** To measure how strongly a trained model depends on the assignment, we replace the matched supports with shuffled ones, $S_{\text{shuf}} = \{(x_i,\, y_{\sigma(i)})\}_{i=1}^{K}$, where $\sigma$ is a *derangement*, a permutation that leaves no label on its own image and exists for $K \geq 2$, drawn by rejection sampling. Every support label now sits on another support image, and the label multiset of the support set is unchanged. We summarize pairing dependence by the pairing gap

$$\Delta = M_{\text{shuffled}} - M_{\text{matched}}, \tag{2}$$

where $M$ is the task metric (DSC or PSNR) and the support set is resampled per query; a model that depends on the pairing loses accuracy under the derangement and has a negative gap.

**Late unpairing curriculum.** We regulate pairing dependence in training with random unpairing. In an unpairing epoch, every support in every episode has its label replaced by a copy of the label of another support in the same set, chosen at random; the query and its target are unchanged. A label can therefore appear twice in the support set while another is dropped, so the training operator does not preserve the label multiset, while the test-time derangement does. The operation is the same for every model we train; what differs is the schedule $p(e)$, which marks each epoch $e$ as matched ($p(e)$=0) or unpairing ($p(e)$=1). The *late unpairing curriculum* (LUC) keeps $p(e)$=0 for the first $e_0$ epochs and sets $p(e)$=1 afterwards, so the model first learns from matched supports and then finishes training on unpaired ones. We write LUC$_x$, where $x$ is the fraction of training epochs with unpairing: LUC$_{0.5}$ unpairs in the second half of training, and LUC$_{0.2}$ only in the final fifth. LUC$_{0.5}$ is our primary schedule, and LUC without a subscript refers to it when results are reported. The *early unpairing curriculum* (EUC$_x$) runs the same schedule in reverse, unpairing first and ending on matched supports, and serves as a control for timing, since it unpairs for as many epochs as LUC$_x$. Two constant schedules complete the family: $p(e)$=0 throughout, which is conventional paired training, and $p(e)$=1 throughout, the *unpaired* model. Every Neuroverse3D model we train uses one recipe, detailed in Section A, on a mix of five task types from BraTS and three T1 brain MRI cohorts: tumor segmentation, anatomical segmentation, modality translation, inpainting and bias correction.

## 4 Pairing dependence and the failure modes of paired training

**Pairing gap of ICL models.** The derangement is a black-box, inference-only test, so we apply it to four released models to derive their pairing gap (Table 1; per-model protocols in Section A). On the two BraTS segmentation tasks (Menze et al., 2014), the four models differ substantially in how strongly they depend on the pairing. On the whole-tumor segmentation task, the test-time derangement induces a DSC drop of at least $0.11$ (for Neuroverse3D), with the largest drop, $0.63$, observed for the general-purpose SegGPT. For the edema segmentation task, the pairing gap ranges from $-0.35$ to $-0.51$. The gap also differs between tasks for the same model; Neuroverse3D, for example, has a gap of $-0.11$ on whole-tumor segmentation and $-0.35$ on edema segmentation.

**Support-label properties influence query predictions.** The pairing gap measures how much a prediction depends on which label sits on which image; it does not show what this dependence does to the prediction. Under matched supports, each support label is the correct answer for its own image, so a model can take parts of its answer, such as where the tumor lies and how large it is, from the support labels instead of from the query. We examine whether such support-specific properties carry over to the query prediction in the Neuroverse3D model trained with matched supports only ($p$=0),

Table 1: **Pairing gap in four released models on two BraTS tasks.** Each model is run with its own protocol using released weights (inference only); we report DSC on matched supports and the pairing gap, defined as the difference between shuffled and matched DSC.

| | | Whole tumor | | Edema | |
|---|---|---|---|---|---|
| Backbone | Fusion | matched | gap | matched | gap |
| *2D* | | | | | |
| UniverSeg (Butoi et al., 2023) | averaging | 0.731 | $-0.41$ | 0.508 | $-0.36$ |
| SegGPT (Wang et al., 2023b) | feature ensemble | 0.713 | $-0.63$ | 0.510 | $-0.51$ |
| *3D* | | | | | |
| Neuroverse3D (Hu et al., 2025) | mean-over-$K$ | 0.846 | $-0.11$ | 0.593 | $-0.35$ |
| Medverse (Hu et al., 2026) | cross-attention | 0.824 | $-0.25$ | 0.654 | $-0.36$ |

the *paired model*. Using 60 BraTS validation subjects with $K{=}4$ supports, we first inspect this behavior with real, unaltered support sets and then test it directly through controlled perturbations of the support labels (Figure 3).

With real, unaltered support sets, we observe that the model tends to segment regions that are absent in the query image. On average, the model generates $2.40$ such extra regions per case (Figure 3a). These regions are not randomly distributed: they tend to lie where the support labels carry tumor, and simply deleting these extra regions from the segmentation would raise the DSC by $0.041$, which indicates that the support-associated behavior accounts for a measurable fraction of the model's error. Moreover, we observe that the predicted tumor size also follows the support labels: after accounting for the true tumor volume, larger support labels are associated with larger predicted tumor volumes (Figure 3b). Section C gives details on counting the extra regions and on the volume analysis.

**Controlled perturbations of support-label properties.** Given the above evidence, we next causally test whether properties of the support labels influence the query prediction. We perturb the support labels while leaving the query and support images unchanged, using two interventions. First, we shift every support label by $8$ voxels along each of two axes, as a mis-registration would. The model's predicted tumor centroid moves $0.67$ voxels along the shift (Figure 3c). This perturbation also reduces whole-tumor DSC by $0.035$ (Figure 10 in Section D shows this cost next to replaced and eroded support labels), showing that the transferred spatial bias has a measurable effect on segmentation accuracy. Next, we inject one synthetic mask into every support label, at a place where the query has no tumor. This intervention makes the model predict most of the injected region as tumor: the fraction of mask voxels predicted as tumor rises by $0.846$ (Figure 8 in Section C shows an example). In every one of these tests, the prediction of the model follows the support labels rather than the query.

## 5 LUC REDUCES THE DEPENDENCE AND THE FAILURE MODES

Paired training gives a model no reason to doubt a single support label, because every support label belongs to its own image. We want a model that still learns the task from its supports but no longer takes its answer from individual support labels. Random unpairing (Section 3) supplies that reason: once each support carries another support's label, a single label is no longer a reliable answer, while the labels as a set still show which output is requested. LUC applies it late, after a phase of matched training in which the model learns from correct pairs how images map to labels.

### 5.1 PAIRING GAP AND MATCHED-SUPPORT PERFORMANCE

**LUC closes the pairing gap.** Under the test-time derangement, LUC predictions barely depend on which label sits on which support image. On the BraTS whole-tumor segmentation task, the gap shrinks from $-0.184$ for the paired model to $-0.008$ for $\text{LUC}_{0.5}$ (Figure 4), and on all five trained task types it is near zero for both LUC models (Table 2). The reduction extends beyond BraTS and beyond one backbone. On three anatomical cohorts spanning children in ABCD (Casey et al., 2018), older adults in ADNI (Jack Jr et al., 2008) and a mixed population in OASIS-3 (LaMontagne et al.,

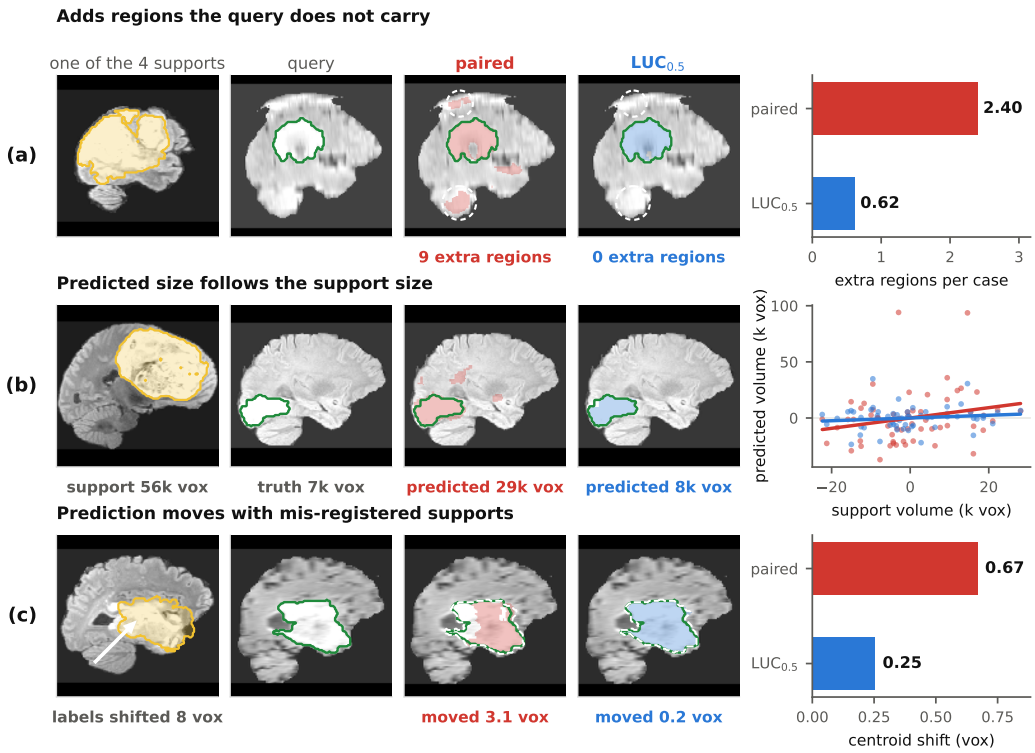


Figure 3: **The influence of support labels on query predictions.** Green outline: ground truth label; filled: model's prediction. The first column shows one of the $K{=}4$ support images, with its label in yellow. Each row shows a prediction failure, with the right-hand panel reporting statistics of the artifact over $n{=}60$ validation subjects. (a) Based on real, unaltered supports, the paired model predicts additional regions that do not overlap the ground truth (white dashed circles). (b) Based on real, unaltered supports, volumes of the tumor predicted by the paired model correlate with tumor size in the support. $\text{LUC}_{0.5}$ removes this spurious correlation. (c) Shifting every support label by $8$ voxels along each of two axes (white arrow in the first column) causes the paired model's prediction to shift in the same direction; in the prediction panels, the dashed outline is the prediction with the original supports and the filled region the prediction with the shifted supports.

Table 2: **Matched-support score and pairing gap on five task types.** The models differ only in the unpairing schedule $p(e)$. Green: an LUC cell with a near-zero gap and a matched-support score at least equal to that of the paired model; red: the paired model's gap. Per-task detail is in Section D.

| | paired ($p{=}0$) | | $\text{LUC}_{0.5}$ | | $\text{LUC}_{0.2}$ | |
|---|---|---|---|---|---|---|
| **Task (metric)** | matched | gap | matched | gap | matched | gap |
| Segmentation (DSC ↑) | 0.733 | −0.184 | 0.857 | −0.008 | 0.878 | −0.003 |
| Translation (PSNR ↑) | 22.41 | −1.22 | 22.56 | −0.03 | 22.70 | −0.08 |
| Anatomical seg. (DSC ↑) | 0.704 | −0.035 | 0.806 | −0.001 | 0.791 | −0.007 |
| Inpainting (PSNR ↑) | 17.52 | −2.27 | 17.76 | +0.05 | 17.88 | −0.02 |
| Bias corr. (PSNR ↑) | 26.59 | −4.20 | 27.45 | −0.24 | 26.34 | −0.18 |

2019), $\text{LUC}_{0.5}$ leaves a gap within $0.004$ DSC of zero (Table 7 in Section D gives the gap for each cohort). When applied to 2D UniverSeg trained from scratch on BraTS multi-region segmentation, LUC reduces the gap from $-0.222$ to about zero (Table 8 in Section D compares the two backbones).

**LUC improves matched-support performance.** Importantly, the reduction in pairing dependence is not achieved by trading off performance under correctly paired supports; LUC instead improves matched-support performance. Across all five trained task types, LUC maintains or improves

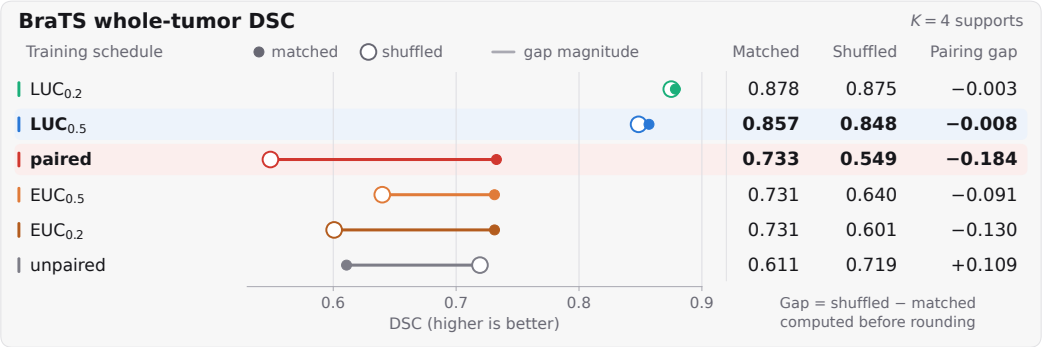


Figure 4: **Matched-support and shuffled-support DSC on BraTS whole-tumor segmentation.** Filled marker: DSC evaluated on matched supports; open marker: DSC evaluated with the shuffled support labels (test-time derangement); the line between them is the pairing gap. LUC combines a near-zero pairing gap with high matched-support performance. Reversing the schedule with EUC retains a larger gap, while unpairing in every epoch (the unpaired model) lowers matched-support performance.

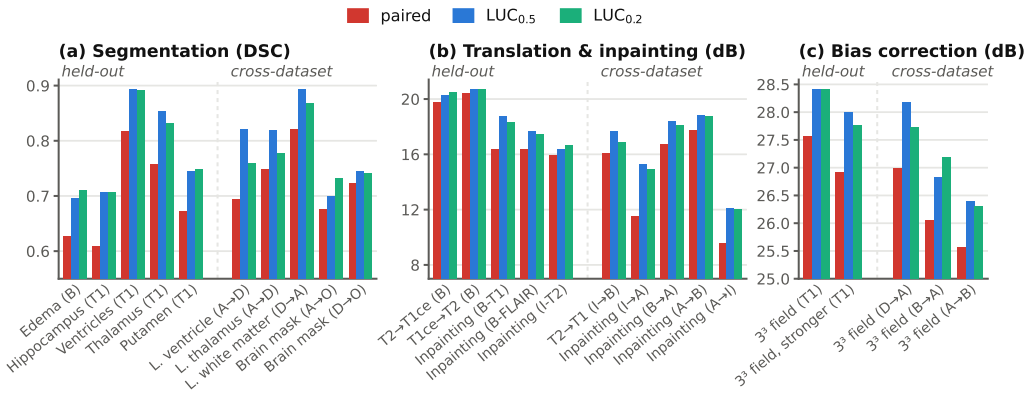


Figure 5: **Performance on tasks outside training.** Matched-support score on held-out tasks and on cross-dataset episodes, in which the supports and the query come from different datasets. Held-out segmentation tasks use new combinations of anatomical structures that are segmented individually during training, while the bias-field-correction task in (c) uses a coarser $3^3$ bias-field grid than in training. Both LUC models score significantly above the paired model on every task shown (subject-level Wilcoxon, $p<0.05$); Table 10 in Section G lists all such tasks. Datasets in parentheses: B: BraTS, A: ABCD, D: ADNI, O: OASIS-3, I: IXI; T1: the three T1 cohorts.

matched-support performance relative to the paired model (Table 2). On the primary whole-tumor segmentation task, $\text{LUC}_{0.5}$ increases the matched-support DSC from $0.733$ to $0.857$ (subject-level Wilcoxon $p=2.4\times10^{-5}$). The improvement also extends across populations, with DSC on ADNI increasing from $0.466$ to $0.737$, and UniverSeg keeps its matched-support DSC.

The gain also holds outside the training distribution. Held-out tasks change the segmentation target, the translation direction or the inpainting setting to ones that never occur in training, and cross-dataset episodes take the supports and the query from different datasets, as when a support set was collected on another scanner or population, whereas every training episode draws both from one dataset. On $46$ of $53$ such tasks, both LUC models have a significantly higher matched-support score than the paired model, with near-zero gaps. Figure 5 shows $25$ of these tasks, and Table 10 in Section G lists all $53$. Notably, for a held-out bias-correction task with a coarser bias field than those seen during training, the paired model barely corrects the input while both LUC models do (Figure 5c).

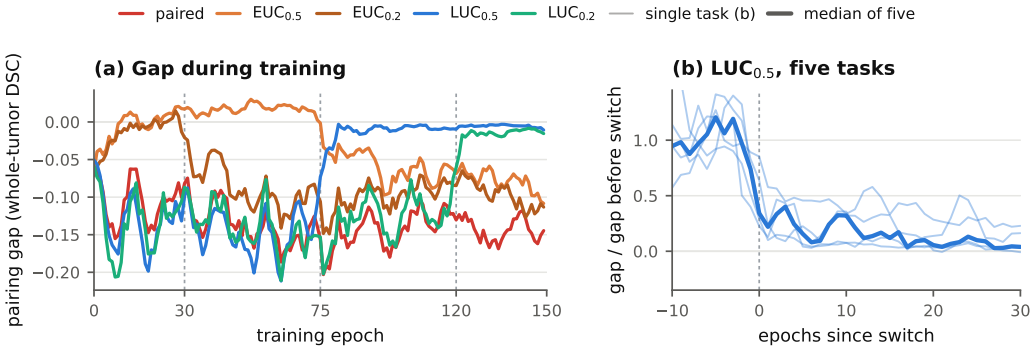


Figure 6: **Pairing gap during training, and how fast it closes after the switch.** Validation gap logged at every epoch. (a) Whole-tumor gap, five-epoch moving average; dotted lines mark the schedule switches. (b) Gap on each of five tasks (BraTS whole tumor, ABCD, ADNI, OASIS-3, T2→FLAIR), divided by its mean over the ten epochs before the switch, for $\mathrm{LUC}_{0.5}$; three-epoch moving average, thin lines for individual tasks and the thick line for their median. Figure 12 in Section E shows that $\mathrm{LUC}_{0.2}$ behaves the same way.

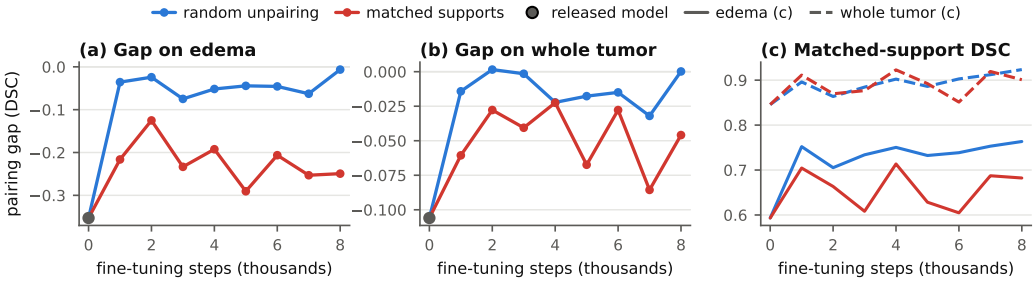


Figure 7: **Fine-tuning the released Neuroverse3D of Table 1, with random unpairing or with matched supports.** Pairing gap on edema (a) and on whole tumor (b), and matched-support DSC (c), against fine-tuning steps. Step 0 is the released model; every evaluation uses the same queries, supports and derangements as Table 1.

**The timing of unpaired training matters.** Unpairing in every epoch does not enjoy the same combination of low pairing dependence and high matched-support performance as LUC: the unpaired model reaches a matched-support DSC of only $0.611$, against $0.733$ for the paired model (Figure 4). On the other hand, EUC (trained on unpaired supports first and matched supports last) cannot fully close the gap; e.g., $\mathrm{EUC}_{0.5}$ and $\mathrm{EUC}_{0.2}$ keep whole-tumor gaps of $-0.091$ and $-0.130$ (Figure 4). The gap logged at every training epoch (Figure 6) shows that $\mathrm{LUC}_{0.5}$ reduces the pairing gap within a few epochs after switching to unpaired supports on all five tasks tracked during training, whereas both EUC models regain their gap once training returns to matched supports. A short final phase of unpairing is therefore enough, and it adds no parameters, training steps, labels or inference cost.

Therefore, for a released model the matched phase has already been run by its authors, so a short phase of random unpairing on top of it completes an LUC schedule. We test this by fine-tuning the released Neuroverse3D on our data mix with random unpairing. As a control, we separately fine-tune the model with matched supports for the same number of steps (Figure 7). After $1000$ steps of fine-tuning, a small fraction of one training run of our recipe, the gap for edema segmentation shrinks from $-0.353$ to $-0.036$ and stays near zero, while the control keeps a large gap; matched-support DSC rises in both arms (Section D gives the values over the whole fine-tuning run).

### 5.2 LUC REDUCES FAILURE MODES

$\text{LUC}_{0.5}$ reduces each failure mode of Section 4, and Figure 3 compares $\text{LUC}_{0.5}$ and the paired model on the same cases and supports. With real support sets, $\text{LUC}_{0.5}$ adds fewer regions that the query does not have ($0.62$ per case against $2.40$). Removing these extra regions hardly impacts the DSC of $\text{LUC}_{0.5}$ ($0.007$, against $0.041$ for the paired model). Moreover, with $\text{LUC}_{0.5}$ the predicted tumor size hardly follows the tumor size in the support labels. Under the controlled perturbations, shifting the support labels barely moves its prediction ($0.25$ voxels for $\text{LUC}_{0.5}$ against $0.67$ for the paired model), and the shift also costs $\text{LUC}_{0.5}$ only $0.001$ DSC, against $0.035$ for the paired model. Lastly, an injected mask raises the predicted fraction of the mask by only $0.095$, against $0.846$ for the paired model (Section F gives the other models). The timing of the unpairing matters here too: $\text{EUC}_{0.5}$, which ends on matched supports, still predicts most of the injected mask ($0.817$; Figure 11 in Section E). Support-specific properties thus carry over far less to the query predictions of $\text{LUC}_{0.5}$.

## 6 DISCUSSION

**The derangement as a test for other models.** Any in-context learner that conditions on a set of examples (Brown et al., 2020; Wang et al., 2023b) draws on two sources of information: the correspondence within each example, and the examples as a set. Deranging the support labels disrupts the first and holds the second fixed, and in the paired model it costs more DSC than replacing the labels with other patients' labels, which also changes which labels the set contains (Section B). The derangement needs no new labels or held-out split and runs on a released model as it is. Pairing dependence appears in all four released models, across 2D and 3D, four ways of fusing the supports and a general-purpose model, and its strength differs between models and tasks (Table 1), so the derangement can be added to the evaluation of any visual ICL model.

**Where the pairing dependence comes from, and how to regulate it.** Under matched supports every label is correct for its own image, so relying on the pairing works throughout paired training. The dependence follows the current phase of training: the gap closes within a few epochs of a switch to unpaired supports and comes back at a similar rate once training returns to matched supports (Figure 6). As in curriculum learning (Bengio et al., 2009), the order of training matters, and for the gap it matters more than the amount of unpairing: $\text{LUC}_{0.2}$, which unpairs only in the final fifth, has a gap near zero, while $\text{EUC}_{0.5}$, which unpairs for the whole first half, keeps a gap (Figure 4). A practitioner therefore sets the dependence by choosing when unpairing starts, and a released model needs only a short final phase of unpairing (Figure 7).

**Support-label errors in medical imaging.** Support labels in medical in-context learning are few and drawn by hand (Karimi et al., 2020), and a label can end up out of register with its scan. In the paired model, mis-registered support labels move the prediction and lower its accuracy (Section 4). LUC reduces reliance on individual support labels while maintaining or improving matched-support performance: the same shift barely moves its prediction and costs almost no accuracy (Section 5.2).

## 7 CONCLUSION

Visual in-context learners infer a task from support pairs, and we asked how strongly their predictions depend on the individual image–label pairings. Deranging the support labels at test time leaves the label multiset unchanged and moves only the pairing, and the resulting pairing gap shows that this dependence varies widely across four released models. The late unpairing curriculum, random unpairing scheduled late in training, substantially reduces the dependence and the failure modes of paired training while maintaining or improving matched-support performance across all evaluated task types, and a short unpairing phase shrinks the gap of a released model. The reduction in the gap holds across three populations, on held-out tasks and on cross-dataset episodes. With the same number of unpairing epochs, the schedule that ends on unpaired supports has the gap closer to zero, so the phase that ends training sets how much the prediction depends on the pairing. The derangement diagnoses pairing dependence, and the schedule $p(e)$ controls it.

## ETHICS STATEMENT

This work uses public, de-identified brain MRI datasets and collects no new data from human subjects. The failure modes we report, a prediction that follows the support labels rather than the query, matter for clinical use, and the training schedule we describe reduces them without extra annotation.

## REPRODUCIBILITY STATEMENT

Every Neuroverse3D model we trained uses one recipe (Section 3) in which only the unpairing schedule $p(e)$ varies, and the model, learning-rate schedule, data, task mix, optimizer, gradient clip, batch size, steps per epoch, seed, epoch budget and hardware were held fixed across runs. Section A gives the training protocol, the unpairing schedules, the evaluation protocol for the released models, and the definition of every probe, together with the subject-level evaluation and its bootstrap confidence intervals. Code to reproduce our results is available at `https://anonymous.4open.science/r/Unpair-Med-vICL`.

## AI USE STATEMENT

Generative AI tools helped the authors draft and polish the manuscript prose and write the figure-plotting and analysis-formatting code, under close author direction. The authors designed the experiments and produced the results. Every experiment, number, table, and figure was produced, checked, and verified by the authors, who take full responsibility for the content.

## REFERENCES

Alessandro Achille, Matteo Rovere, and Stefano Soatto. Critical learning periods in deep networks. In *International conference on learning representations*, 2018.

Yutong Bai, Xinyang Geng, Karttikeya Mangalam, Amir Bar, Alan L Yuille, Trevor Darrell, Jitendra Malik, and Alexei A Efros. Sequential modeling enables scalable learning for large vision models. In *2024 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 22861–22872. IEEE, 2024.

Ujjwal Baid, Satyam Ghodasara, Suyash Mohan, Michel Bilello, Evan Calabrese, Errol Colak, Keyvan Farahani, Jayashree Kalpathy-Cramer, Felipe C Kitamura, Sarthak Pati, et al. The RSNA-ASNR-MICCAI BraTS 2021 benchmark on brain tumor segmentation and radiogenomic classification. *arXiv preprint arXiv:2107.02314*, 2021.

Spyridon Bakas, Hamed Akbari, Aristeidis Sotiras, Michel Bilello, Martin Rozycki, Justin S Kirby, John B Freymann, Keyvan Farahani, and Christos Davatzikos. Advancing The Cancer Genome Atlas glioma MRI collections with expert segmentation labels and radiomic features. *Scientific data*, 4(1):170117, 2017.

Amir Bar, Yossi Gandelsman, Trevor Darrell, Amir Globerson, and Alexei Efros. Visual prompting via image inpainting. *Advances in neural information processing systems*, 35:25005–25017, 2022.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pp. 41–48, 2009.

Benjamin Billot, Douglas N Greve, Oula Puonti, Axel Thielscher, Koen Van Leemput, Bruce Fischl, Adrian V Dalca, Juan Eugenio Iglesias, et al. SynthSeg: Segmentation of brain MRI scans of any contrast and resolution without retraining. *Medical image analysis*, 86:102789, 2023.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Victor Ion Butoi, Jose Javier Gonzalez Ortiz, Tianyu Ma, Mert R Sabuncu, John Guttag, and Adrian V Dalca. UniverSeg: Universal medical image segmentation. In *2023 IEEE/CVF international conference on computer vision (ICCV)*, pp. 21381–21394. IEEE, 2023.

Betty Jo Casey, Tariq Cannonier, May I Conley, Alexandra O Cohen, Deanna M Barch, Mary M Heitzeg, Mary E Soules, Theresa Teslovich, Danielle V Dellarco, Hugh Garavan, et al. The adolescent brain cognitive development (ABCD) study: imaging acquisition across 21 sites. *Developmental cognitive neuroscience*, 32:43–54, 2018.

Stephanie Chan, Adam Santoro, Andrew Lampinen, Jane Wang, Aaditya Singh, Pierre Richemond, James McClelland, and Felix Hill. Data distributional properties drive emergent in-context learning in transformers. *Advances in neural information processing systems*, 35:18878–18891, 2022.

Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3D U-Net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, pp. 424–432. Springer, 2016.

Steffen Czolbe and Adrian V Dalca. Neuralizer: General neuroimage analysis without re-training. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6217–6230. IEEE, 2023.

Yunhe Gao, Di Liu, Zhuowei Li, Yunsheng Li, Dongdong Chen, Mu Zhou, and Dimitris N Metaxas. Show and segment: Universal medical image segmentation via in-context learning. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20830–20840. IEEE, 2025.

Alex Graves, Marc G Bellemare, Jacob Menick, Remi Munos, and Koray Kavukcuoglu. Automated curriculum learning for neural networks. In *international conference on machine learning*, pp. 1311–1320. Pmlr, 2017.

Guy Hacohen and Daphna Weinshall. On the power of curriculum learning in training deep networks. In *International conference on machine learning*, pp. 2535–2544. PMLR, 2019.

Jiesi Hu, Hanyang Peng, Yanwu Yang, Xutao Guo, Yang Shang, Pengcheng Shi, Chenfei Ye, and Ting Ma. Neuroverse3D: Developing in-context learning universal model for neuroimaging in 3D. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 21721–21731. IEEE, 2025.

Jiesi Hu, Jianfeng Cao, Yanwu Yang, Chenfei Ye, Yixuan Zhang, Hanyang Peng, and Ting Ma. Medverse: A universal model for full-resolution 3D medical image segmentation, transformation and enhancement. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 4869–4877, 2026.

Clifford R Jack Jr, Matt A Bernstein, Nick C Fox, Paul Thompson, Gene Alexander, Danielle Harvey, Bret Borowski, Paula J Britson, Jennifer L. Whitwell, Chadwick Ward, et al. The Alzheimer's disease neuroimaging initiative (ADNI): MRI methods. *Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 27(4): 685–691, 2008.

Davood Karimi, Haoran Dou, Simon K Warfield, and Ali Gholipour. Deep learning with noisy labels: Exploring techniques and remedies in medical image analysis. *Medical image analysis*, 65:101759, 2020.

Donggyun Kim, Jinwoo Kim, Seongwoong Cho, Chong Luo, and Seunghoon Hong. Universal few-shot learning of dense prediction tasks with visual token matching. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=88nT0j5jAn.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *2023 IEEE/CVF international conference on computer vision (ICCV)*, pp. 3992–4003. IEEE, 2023.

Jannik Kossen, Yarin Gal, and Tom Rainforth. In-context learning learns label relationships but is not conventional learning. In *International Conference on Learning Representations*, volume 2024, pp. 3209–3280, 2024.

Pamela J LaMontagne, Tammie LS Benzinger, John C Morris, Sarah Keefe, Russ Hornbeck, Chengjie Xiong, Elizabeth Grant, Jason Hassenstab, Krista Moulder, Andrei G Vlassenko, et al. OASIS-3: longitudinal neuroimaging, clinical, and cognitive dataset for normal aging and Alzheimer disease. *medrxiv*, 2019.

Jie Liu, Yixiao Zhang, Jie-Neng Chen, Junfei Xiao, Yongyi Lu, Bennett A Landman, Yixuan Yuan, Alan Yuille, Yucheng Tang, and Zongwei Zhou. CLIP-driven universal model for organ segmentation and tumor detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 21152–21164, 2023.

Yang Liu, Muzhi Zhu, Hengtao Li, Hao Chen, Xinlong Wang, and Chunhua Shen. Matcher: Segment anything with one shot using all-purpose feature matching. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=yzRXdhk2he.

Jun Ma, Yuting He, Feifei Li, Lin Han, Chenyu You, and Bo Wang. Segment anything in medical images. *Nature communications*, 15(1):654, 2024.

Bjoern H Menze, Andras Jakab, Stefan Bauer, Jayashree Kalpathy-Cramer, Keyvan Farahani, Justin Kirby, Yuliya Burren, Nicole Porz, Johannes Slotboom, Roland Wiest, et al. The multimodal brain tumor image segmentation benchmark (BRATS). *IEEE transactions on medical imaging*, 34(10):1993–2024, 2014.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? In *Proceedings of the 2022 conference on empirical methods in natural language processing*, pp. 11048–11064, 2022.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, et al. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*, 2022.

Cheng Ouyang, Carlo Biffi, Chen Chen, Turkay Kart, Huaqi Qiu, and Daniel Rueckert. Self-supervision with superpixels: Training few-shot medical image segmentation without annotation. In *European conference on computer vision*, pp. 762–780. Springer, 2020.

Cheng Ouyang, Chen Chen, Surui Li, Zeju Li, Chen Qin, Wenjia Bai, and Daniel Rueckert. Causality-inspired single-source domain generalization for medical image segmentation. *IEEE Transactions on Medical Imaging*, 42(4):1095–1106, 2022.

Jane Pan, Tianyu Gao, Howard Chen, and Danqi Chen. What in-context learning "learns" in-context: Disentangling task recognition and task learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 8298–8319, 2023.

Janarthanan Rajendran, Alexander Irpan, and Eric Jang. Meta-learning requires meta-augmentation. *Advances in Neural Information Processing Systems*, 33:5705–5715, 2020.

Marianne Rakic, Hallee E Wong, Jose Javier Gonzalez Ortiz, Beth A Cimini, John V Guttag, and Adrian V Dalca. Tyche: Stochastic in-context learning for medical image segmentation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11159–11173. IEEE, 2024.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Dianmo Sheng, Dongdong Chen, Zhentao Tan, Qiankun Liu, Qi Chu, Tao Gong, Bin Liu, Jing Han, Wenbin Tu, Shengwei Xu, et al. UNICL-SAM: Uncertainty-driven in-context segmentation with part prototype discovery. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20201–20211. IEEE, 2025.

Aaditya Singh, Stephanie Chan, Ted Moskovitz, Erin Grant, Andrew Saxe, and Felix Hill. The transient nature of emergent in-context learning in transformers. *Advances in neural information processing systems*, 36:27801–27819, 2023.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017.

Petru Soviany, Radu Tudor Ionescu, Paolo Rota, and Nicu Sebe. Curriculum learning: A survey. *International Journal of Computer Vision*, 130(6):1526–1565, 2022.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29, 2016.

Haoyu Wang, Sizheng Guo, Jin Ye, Zhongying Deng, Junlong Cheng, Tianbin Li, Jianpin Chen, Yanzhou Su, Ziyan Huang, Yiqing Shen, et al. SAM-Med3D: a vision foundation model for general-purpose segmentation on volumetric medical images. *IEEE Transactions on Neural Networks and Learning Systems*, 2025.

Xinlong Wang, Wen Wang, Yue Cao, Chunhua Shen, and Tiejun Huang. Images speak in images: A generalist painter for in-context visual learning. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6830–6839. IEEE, 2023a.

Xinlong Wang, Xiaosong Zhang, Yue Cao, Wen Wang, Chunhua Shen, and Tiejun Huang. SegGPT: Towards segmenting everything in context. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1130–1140. IEEE, 2023b.

Jerry Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, et al. Larger language models do in-context learning differently. *arXiv preprint arXiv:2303.03846*, 2023.

Xiaoxia Wu, Ethan Dyer, and Behnam Neyshabur. When do curricula work? In *International Conference on Learning Representations*, 2021.

Mingzhang Yin, George Tucker, Mingyuan Zhou, Sergey Levine, and Chelsea Finn. Meta-learning without memorization. In *International Conference on Learning Representations*, 2020.

Renrui Zhang, Zhengkai Jiang, Ziyu Guo, Shilin Yan, Junting Pan, Hao Dong, Yu Qiao, Gao Peng, and Hongsheng Li. Personalize segment anything model with one shot. In *International Conference on Learning Representations*, volume 2024, pp. 18250–18279, 2024.

Yuanhan Zhang, Kaiyang Zhou, and Ziwei Liu. What makes good examples for visual in-context learning? *Advances in Neural Information Processing Systems*, 36:17773–17794, 2023.

# A IMPLEMENTATION AND EVALUATION DETAILS

This appendix describes the controlled training family, the data and preprocessing, the support-set operators, the evaluation of the released models, and the statistical procedures used in Section 3.

### ARCHITECTURE AND TRAINING RECIPE

The backbone is Neuroverse3D (Hu et al., 2025), a five-stage 3D U-Net (Ronneberger et al., 2015; Çiçek et al., 2016) with channels $[32, 64, 128, 256, 512]$ and 70.9M parameters. It receives a query image and $K$ support (image, label) pairs and predicts a dense output for the query. The six models of the controlled family use Adam with learning rate $10^{-4}$ and no weight decay, loss $50 \times \mathcal{L}_{\text{smooth-}L_3\text{-}L_1}$, gradient clipping to norm $0.25$, batch size 1, fp32 precision, $2000$ steps per epoch, and $150$ epochs with seed $42$. Flip, Sobel filtering, and the GIN intensity augmentation of Ouyang et al. (2022) are each applied at a rate of $0.05$. Training uses three supports; evaluation uses $K{=}4$ unless a sweep is stated. All settings in Table 3, including the learning-rate schedule and hardware, are fixed across the family; only $p(e)$ varies.

Table 3: **Training recipe for the controlled Neuroverse3D family.** All six models share these settings and differ only in the unpairing schedule $p(e)$.

| Ingredient | Value (fixed across the family) |
|---|---|
| Architecture | Neuroverse3D, channels $[32, 64, 128, 256, 512]$, $\sim$71M params |
| Data mix | BraTS 2021 + three T1 datasets (ABCD, ADNI, OASIS-3) |
| Tasks | segmentation, translation, anatomical seg., inpainting, bias correction |
| Optimizer | Adam, lr $10^{-4}$, no weight decay |
| Learning-rate schedule | held fixed across the family |
| Loss | $50 \times \mathcal{L}_{\text{smooth-}L_3\text{-}L_1}$ |
| Gradient clip / batch / steps-per-epoch | 0.25 (norm) / 1 / 2000 |
| Precision / support-set size | fp32 / 3 (train), $K{=}4$ (eval) |
| Augmentation | flip / Sobel / GIN-contrast, each at rate $0.05$ |
| Seed / epoch budget / hardware | 42 / 150 / uniform GPU |
| **Varies** | **unpairing schedule $p(e)$ only** |

### RANDOM UNPAIRING

Each epoch is either matched ($p(e){=}0$), with every support keeping its own label, or unpairing ($p(e){=}1$). In an unpairing epoch every support keeps its image and receives a copy of the label of a uniformly chosen other support. All replacement labels come from the same episode, so one label can appear twice while another is dropped: random unpairing does not preserve the label multiset, whereas the test-time derangement below does. The query and its target are never changed, so the loss is always computed against the correct answer. A schedule is summarized by $x$, the fraction of the $E{=}150$ training epochs with $p(e){=}1$.

### THE MODEL FAMILY

The family contains six models (Table 4): the paired model ($p(e){=}0$ throughout), the unpaired model ($p(e){=}1$ throughout), and two time-reversed pairs. $\text{EUC}_{0.5}$ and $\text{LUC}_{0.5}$ switch at epoch 75; $\text{EUC}_{0.2}$ and $\text{LUC}_{0.2}$ place 30 unpairing epochs at the beginning or at the end of training. Within each pair the number of unpairing epochs is the same, so the reversal changes only when the unpairing happens.

### DATA AND PREPROCESSING

Training uses BraTS 2021 (Menze et al., 2014; Bakas et al., 2017; Baid et al., 2021), with co-registered T1, T1ce, T2 and FLAIR images and nested ET/TC/WT labels, together with three T1 datasets, ABCD (Casey et al., 2018), ADNI (Jack Jr et al., 2008) and OASIS-3 (LaMontagne et al., 2019), which carry SynthSeg anatomical labels (Billot et al., 2023). Preprocessing clips intensities to the $[0.5\%, 99.5\%]$ percentiles, applies z-score normalization, then crops, pads and resizes each volume to $128^3$; images receive a final min–max normalization. Labels use the crop coordinates

Table 4: **The six models.** All share the recipe of Table 3 and differ only in $p(e)$; $x$ is the fraction of epochs with unpairing, and "mirror" names the time-reversed partner.

| model | $p(e)$ | $x$ | mirror |
|---|---|---|---|
| paired | 0 for all $e$ | 0 | none |
| unpaired | 1 for all $e$ | 1 | none |
| $\text{EUC}_{0.5}$ | 1 for $e<75$, else 0 | 0.5 | $\text{LUC}_{0.5}$ |
| $\text{LUC}_{0.5}$ | 0 for $e<75$, else 1 | 0.5 | $\text{EUC}_{0.5}$ |
| $\text{EUC}_{0.2}$ | 1 for $e<30$, else 0 | 0.2 | $\text{LUC}_{0.2}$ |
| $\text{LUC}_{0.2}$ | 0 for $e<120$, else 1 | 0.2 | $\text{EUC}_{0.2}$ |

of their images, so every support image and its label share one spatial frame, and labels are never intensity-normalized. We use a $90/10$ train/validation split per dataset, and all reported numbers are computed on held-out validation subjects. Whole-tumor labels follow the BraTS convention (Menze et al., 2014).

### Released models and fine-tuning

Table 1 evaluates each released model as distributed, without fine-tuning. Queries are FLAIR images of BraTS validation subjects, and supports are drawn from the training split. Neuroverse3D and Medverse take 3D volumes with $K=4$ supports, on 20 and 12 validation subjects. UniverSeg and SegGPT take one tumor-containing axial FLAIR slice per subject at $128\times128$, which SegGPT resizes to $448\times448$. UniverSeg uses its default of 16 supports on 125 validation slices (123 with edema), and SegGPT uses $K=4$ on 24 slices. Each query gets one support draw and one derangement, shared by its matched and shuffled runs. The released Neuroverse3D is also fine-tuned on our data mix at learning rate $10^{-5}$, with random unpairing at $p=1$ or with matched supports. The fine-tuned model of Figure 7 is scored every 1000 steps on the same queries, supports and derangements as the released model. This fine-tuning is separate from the controlled family.

### The derangement and the support-set size

The test-time operator draws a permutation $\sigma$ of the $K$ labels by rejection sampling until $\sigma(i) \neq i$ for all $i$, so no label stays on its own image. It leaves the query, the support images and the label multiset unchanged. A new support set is drawn for each query and held fixed across its matched and shuffled runs. The pairing gap is

$$\Delta = M_{\text{shuffled}} - M_{\text{matched}},$$

where $M$ is DSC or PSNR; a negative gap means the matched supports score higher, a positive gap that the shuffled supports do. A derangement exists for $K \geq 2$. At $K=1$ the only permutation is the identity, so matched and shuffled supports are the same input and the gap is exactly zero, a built-in bookkeeping check. The default is $K=4$.

### The injected-mask test

We add a fixed binary sphere to every support label: radius 9 voxels, 3071 voxels in total, centered at grid location $(44, 44, 64)$. We discard any query whose true tumor overlaps the sphere by more than $2\%$, so the sphere is never a correct answer. The injected-mask response is the change in the fraction of sphere voxels predicted as foreground,

$$R = P(\text{pred} > 0.5 \mid \text{sphere})_{\text{after}} - P(\text{pred} > 0.5 \mid \text{sphere})_{\text{before}},$$

averaged over $n=25$ subjects. A model that reproduces the support labels predicts the sphere and has a large response; a model that reads the answer from the query has a response near zero.

### Subject-level evaluation

Metrics are aggregated to one value per subject before any test, so no subject contributes more than once. Unless stated otherwise, confidence intervals use a subject-level bootstrap with $5{,}000$ resamples; the support-volume regression of Section 4 uses $10{,}000$ resamples of the queries. Within-subject comparisons (matched against shuffled supports, or one model against another) use the

Wilcoxon signed-rank test, with Benjamini–Hochberg FDR control within each task family. The multi-dataset evaluation uses $n$=30 subjects for segmentation, $n$=20 for translation and $n$=12 for anatomical segmentation; the failure-mode analysis uses $n$=60, and each probe states its own sample size. The pairing gap is a within-model, within-subject contrast, so each model is compared with itself.

### RELEASED CODE

Code to reproduce our results is available at `https://anonymous.4open.science/r/Unpair-Med-vICL`.

## B ADDITIONAL DIAGNOSTICS OF PAIRING DEPENDENCE

This appendix reports label-perturbation tests that place the gap in the assignment of labels to images rather than in the label content.

### PAIRING-SPECIFIC CONTROLS

We evaluate the multi-task models under matched, cross-patient, zeroed, Gaussian-noise and shuffled support labels ($K$=4, $n$=30; Table 5).

The contrast that matters is shuffled against zeroed. With shuffled supports, each support of the paired model ($p$=0) carries another support's label from the same support set. Every one of those labels is real and correct for some image in the set. Moving the labels this way costs more than zeroing them: $0.549$ shuffled $<$ $0.655$ zeroed $<$ $0.661$ cross-patient $<$ $0.733$ matched. A model sensitive to any change in the labels would lose at least as much when the labels are zeroed as when they are moved among real images. The measured order is the reverse, which places the loss in the assignment. Gaussian-noise labels, which destroy the label content, score lower still ($0.461$). We therefore read pairing specificity from the ordering against the zeroed and the cross-patient condition: the paired model depends on which label goes with which image.

The LUC models move little between matched and shuffled supports, $0.857$ against $0.848$ for $\text{LUC}_{0.5}$ and $0.878$ against $0.875$ for $\text{LUC}_{0.2}$, and they hold the highest matched-support DSC in the table. Both EUC models, which end on matched supports, lose DSC under shuffling like the paired model ($0.731$ to $0.640$ and $0.731$ to $0.601$). The unpaired model scores $0.611$ with matched supports and $0.719$ with shuffled supports; it was trained only on unpaired supports, so matched ones are new to it. Across the models in this table, pairing dependence follows the training distribution.

Table 5: **Label-perturbation tests on BraTS segmentation** (subject-level mean DSC, $n$=30). Shuffling changes only the assignment; cross-patient, zeroed and noise labels also change the label content. For the paired model, shuffled supports score below the zeroed and the cross-patient condition; the LUC models move little between matched and shuffled supports and hold the highest matched-support DSC.

| model | matched | cross-patient | zero | noise | shuffled |
|---|---|---|---|---|---|
| paired | 0.733 | 0.661 | 0.655 | 0.461 | **0.549** |
| unpaired | 0.611 | 0.713 | 0.714 | 0.573 | 0.719 |
| $\text{LUC}_{0.5}$ | **0.857** | 0.832 | 0.826 | 0.710 | **0.848** |
| $\text{LUC}_{0.2}$ | **0.878** | 0.878 | 0.879 | 0.808 | **0.875** |
| $\text{EUC}_{0.5}$ | 0.731 | 0.669 | 0.631 | 0.541 | 0.640 |
| $\text{EUC}_{0.2}$ | 0.731 | 0.646 | 0.650 | 0.534 | 0.601 |

## C THE FAILURE MODES IN DETAIL

The measurements of Sections 4 and 5.2 use the paired model and $\text{LUC}_{0.5}$ on the same 60 BraTS validation subjects, with the same real support images and support labels ($K$=4).

**Extra regions.** We count predicted connected components of at least $50$ voxels that overlap the query ground truth by less than $10\%$. The paired model produces $2.40$ per case against $0.62$ for $\text{LUC}_{0.5}$ (subject-level Wilcoxon $p{=}1.3{\times}10^{-4}$), and $62\%$ of its cases carry at least one against $30\%$. Deleting those regions raises matched-support DSC by $0.041$ for the paired model and by $0.007$ for $\text{LUC}_{0.5}$, so about a third of the $0.114$ DSC difference between the two models comes from this one failure mode. We also recorded where those components fall. Of their voxels, $0.35$ lie inside the union of the four support labels for the paired model, against $0.27$ for the union of four labels drawn from patients outside the support set. The extra regions therefore fall more often where the support labels carry tumor.

**Predicted size.** We regressed predicted volume on true volume and mean support volume together, so the support term measures the effect of the support volume after accounting for the true volume. Its coefficient is $0.460$ (95% CI $[0.026, 0.990]$) for the paired model and $0.122$ ($[-0.094, 0.346]$) for $\text{LUC}_{0.5}$, a difference of $0.338$ ($[0.037, 0.720]$) from $10{,}000$ bootstrap resamples of the queries, so larger support labels go with a larger prediction about four times as strongly for the paired model.

**Mis-registered supports.** Shifting every support label by $8$ voxels along each of two axes moves the paired model's predicted centroid $0.67$ voxels along the shift direction, against $0.25$ voxels for $\text{LUC}_{0.5}$ ($p{=}0.022$). Figure 10 in Section D gives the DSC this shift costs, together with replaced and eroded support labels.

### A REAL-SUPPORT EXAMPLE AND THE INJECTED-MASK TEST

The effect appears with real supports and no manipulation (Figure 8, top). On a query whose ground truth is a single tumor, the paired model predicts two disconnected regions. It places one of them where the supports carry tumor and the query does not, while $\text{LUC}_{0.5}$ predicts only the real region. This over-segmentation is systematic: across single-tumor queries the paired model adds a spurious disconnected region in $71\%$ of cases, against $38\%$ for $\text{LUC}_{0.5}$. We then paint a fixed synthetic mask into all $K$ support labels at a location outside the query's tumor, so the only change to the input is that mask (Figure 8, middle; definition in Section A). The paired model reproduces the mask in its own prediction (injected-mask response $+0.846$), while the response of $\text{LUC}_{0.5}$ is $+0.095$. Section F gives the response across the family.

## D COMPLETE RESULTS FOR LUC

**One evaluation pipeline.** The task-level tables of this section (Tables 6 and 7) come from one evaluation that scores all six models with a single pipeline. Matched-support scores and pairing gaps are therefore comparable across every row, and they are the numbers behind Table 2. Throughout, "matched" is the matched-support score and "gap" is the pairing gap, the shuffled-support minus the matched-support score, so a more negative gap means more pairing dependence.

### IN-DISTRIBUTION RESULTS ACROSS THE FIVE TRAINED TASK TYPES

Table 6 gives the matched-support score and the gap for every model and trained task type. $\text{LUC}_{0.5}$ brings the gap near zero on all five task types with matched-support scores at or above the paired model ($0.857$, $22.557$, $0.806$, $17.756$ and $27.447$ against $0.733$, $22.406$, $0.704$, $17.517$ and $26.588$). This is the result of Section 5, read task by task. $\text{LUC}_{0.2}$ also brings the gap near zero on all five task types and reaches $0.878$ on segmentation. The unpaired model loses at most $0.011$ under the derangement on every task type, with matched-support segmentation DSC $0.611$. $\text{LUC}_{0.5}$ keeps matched supports through the first half of training and reaches $0.857$.

**Inpainting and bias correction.** The paired model has its largest gaps on the two restoration tasks, $-2.267$ dB for inpainting and $-4.201$ dB for bias correction. $\text{LUC}_{0.5}$ has $+0.052$ and $-0.242$ dB, and $\text{LUC}_{0.2}$ has $-0.021$ and $-0.182$ dB. As on segmentation and translation, the schedules that end on unpaired supports have the gaps closer to zero.

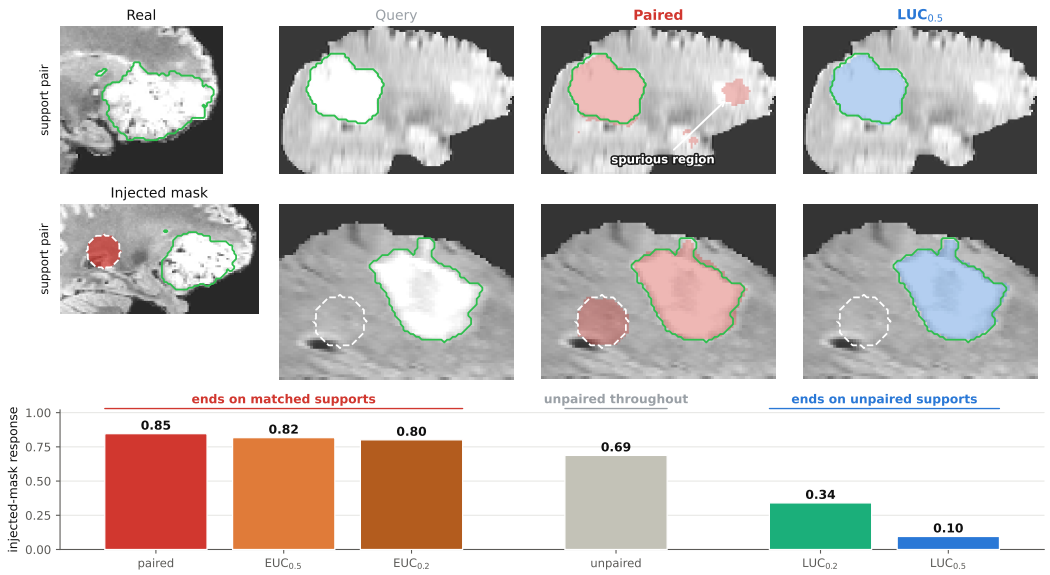


Figure 8: **Predictions with real supports and with a mask painted into the supports.** Each row shows a support pair, the query, and the paired and $\text{LUC}_{0.5}$ predictions. Top: real supports; the query has a single tumor (green). Middle: the injected-mask test, with a fixed synthetic label (red, white dashed ring) painted into every support where the query has no tumor. Bottom: injected-mask response across the models, grouped by schedule.

Table 6: **In-distribution results (matched / gap) for the six models trained under the recipe.** Matched = the score with matched supports, in the metric named in the column header; gap = shuffled − matched, so a more negative gap means more pairing dependence. Green: the LUC models, where the gap is near zero and the matched-support score is at or above the paired model. Red: the paired model's gap. All rows share the one recipe and differ only in $p(e)$.

| model | Seg DSC | Transl PSNR | AnatSeg DSC | Inp PSNR | Bias PSNR |
|---|---|---|---|---|---|
| paired | 0.733 / −0.184 | 22.406 / −1.216 | 0.704 / −0.035 | 17.517 / −2.267 | 26.588 / −4.201 |
| unpaired | 0.611 / +0.109 | 22.146 / +0.042 | 0.685 / +0.005 | 17.327 / −0.011 | 25.958 / +0.159 |
| $\text{EUC}_{0.5}$ | 0.731 / −0.091 | 22.389 / −1.099 | 0.708 / −0.024 | 17.339 / −0.880 | 25.661 / −3.234 |
| $\text{LUC}_{0.5}$ | 0.857 / −0.008 | 22.557 / −0.027 | 0.806 / −0.001 | 17.756 / +0.052 | 27.447 / −0.242 |
| $\text{EUC}_{0.2}$ | 0.731 / −0.130 | 22.486 / −1.340 | 0.694 / −0.025 | 17.651 / −1.872 | 26.516 / −3.759 |
| $\text{LUC}_{0.2}$ | 0.878 / −0.003 | 22.698 / −0.081 | 0.791 / −0.007 | 17.878 / −0.021 | 26.338 / −0.182 |

### Cross-population results

Table 7 splits the anatomical-segmentation result of Table 2 into its three cohorts. On all three cohorts $\text{LUC}_{0.5}$ has a gap within 0.004 of zero and $\text{LUC}_{0.2}$ within 0.016, and both have matched-support DSC above the paired model. On the two cohorts where the paired model has a gap (ABCD and ADNI), $\text{EUC}_{0.5}$ and $\text{EUC}_{0.2}$ have more negative gaps than $\text{LUC}_{0.5}$ and $\text{LUC}_{0.2}$, the same order effect as in Section E. The matched-support gain of $\text{LUC}_{0.5}$ over the paired model is largest on ADNI (0.466 to 0.737), and on OASIS-3 all five models have a gap of zero to three decimals. The pairing gap and its reduction under LUC therefore extend from BraTS whole tumor to anatomical segmentation in children (ABCD) and older adults (ADNI).

### Qualitative multi-task comparison

Figure 9 shows single cases behind the numbers. The top block uses shuffled supports on two tasks. On BraTS whole tumor the paired model returns a partial, displaced mask (DSC ∼0.6), and on T2→FLAIR synthesis it loses tissue detail, while $\text{LUC}_{0.5}$ and $\text{LUC}_{0.2}$ track the target. $\text{EUC}_{0.5}$ and $\text{EUC}_{0.2}$ degrade along with the paired model, and the unpaired model sits between the two.

Table 7: **Pairing gap by cohort (anatomical segmentation, $K{=}4$).** Matched = matched-support DSC; gap = shuffled-support minus matched-support DSC. Green: matched-support DSC under LUC at or above the paired model. Red: the paired model's gap on ABCD. On all three cohorts the LUC models have a gap within $0.016$ of zero and matched-support DSC above paired; $\text{EUC}_{0.5}$ and $\text{EUC}_{0.2}$ (which end on matched supports) have more negative gaps than $\text{LUC}_{0.5}$ and $\text{LUC}_{0.2}$ where the paired model has one.

| | **Paired** ($p{=}0$) | | $\text{LUC}_{0.5}$ | | $\text{LUC}_{0.2}$ | | $\text{EUC}_{0.5}$ | | $\text{EUC}_{0.2}$ | |
|---|---|---|---|---|---|---|---|---|---|---|
| **Cohort** | matched | gap | matched | gap | matched | gap | matched | gap | matched | gap |
| ABCD (children) | 0.859 | −0.074 | 0.868 | +0.002 | 0.871 | −0.006 | 0.848 | −0.049 | 0.857 | −0.048 |
| ADNI (elderly) | 0.466 | −0.030 | 0.737 | −0.004 | 0.702 | −0.016 | 0.504 | −0.024 | 0.438 | −0.026 |
| OASIS-3 (mixed) | 0.785 | 0.000 | 0.813 | 0.000 | 0.800 | 0.000 | 0.774 | 0.000 | 0.788 | 0.000 |

Table 8: **Pairing gap with and without LUC, in a 3D and a 2D backbone.** Each backbone is trained with its own recipe; its two rows differ only in the unpairing schedule $p(e)$. Neuroverse3D: BraTS whole tumor; UniverSeg: BraTS multi-region, mean over WT/TC/ET/edema.

| backbone | training | matched DSC | pairing gap (DSC) |
|---|---|---|---|
| Neuroverse3D (Hu et al., 2025) (3D) | paired | 0.733 | −0.184 |
| | + $\text{LUC}_{0.5}$ | **0.857** | −0.008 |
| UniverSeg (Butoi et al., 2023) (2D) | paired | 0.590 | −0.222 |
| | + $\text{LUC}_{0.5}$ | 0.602 | ≈0.000 |

The bottom block uses matched supports and spans the trained task suite across all four datasets: whole-tumor segmentation, three modality-translation directions, anatomical segmentation (ADNI), inpainting (OASIS-3) and bias-field correction (ABCD). Accuracy there is comparable across the models. $\text{LUC}_{0.5}$ matches the paired model on translation and restoration and fills the inpainting box slightly better. On the two segmentation tasks it drops the spurious whole-tumor region the paired model adds, and it recovers the atrophic ADNI ventricle the paired model under-segments. Tasks outside training are quantified in Section G.

### A SECOND BACKBONE: UNIVERSEG

We trained UniverSeg from scratch on BraTS multi-region segmentation, where the support pair specifies which of WT, TC, ET and edema to segment, and added LUC to its own recipe (Table 8). The gap shrinks from $-0.222$ to $\approx 0$ with matched-support DSC $0.590$ and $0.602$, so LUC carries over to a 2D backbone. Trained from scratch on a single task, UniverSeg shows no gap we could measure.

### LUC ON THE RELEASED NEUROVERSE3D

Figure 7 fine-tunes the released Neuroverse3D with random unpairing and, as a control, with matched supports, following the protocol in Section A. The first $1000$ steps of unpairing, $0.3\%$ of the $300{,}000$ steps in one training run of our recipe, take the edema gap from $-0.353$ to $-0.036$ and the whole-tumor gap from $-0.106$ to $-0.014$. The gaps stay near zero for the remaining $7000$ steps (means over the eight evaluations $-0.043$ on edema and $-0.013$ on whole tumor), while the control keeps an edema gap of $-0.221$ on average, so the drop comes from the unpairing rather than from further training on our data. Matched-support DSC rises in both arms, and on edema it is higher with unpairing at every evaluation ($0.741$ against $0.662$ on average).

### REPLACED, SHIFTED AND ERODED SUPPORT LABELS

The derangement is a controlled test, and we also measured support-set errors that occur in practice (Figure 10). As we replace $0$ to $4$ of the $K{=}4$ support labels with another patient's label, the paired model's BraTS whole-tumor DSC falls from $0.74$ to $0.62$, while $\text{LUC}_{0.5}$ goes from $0.85$ to $0.83$.

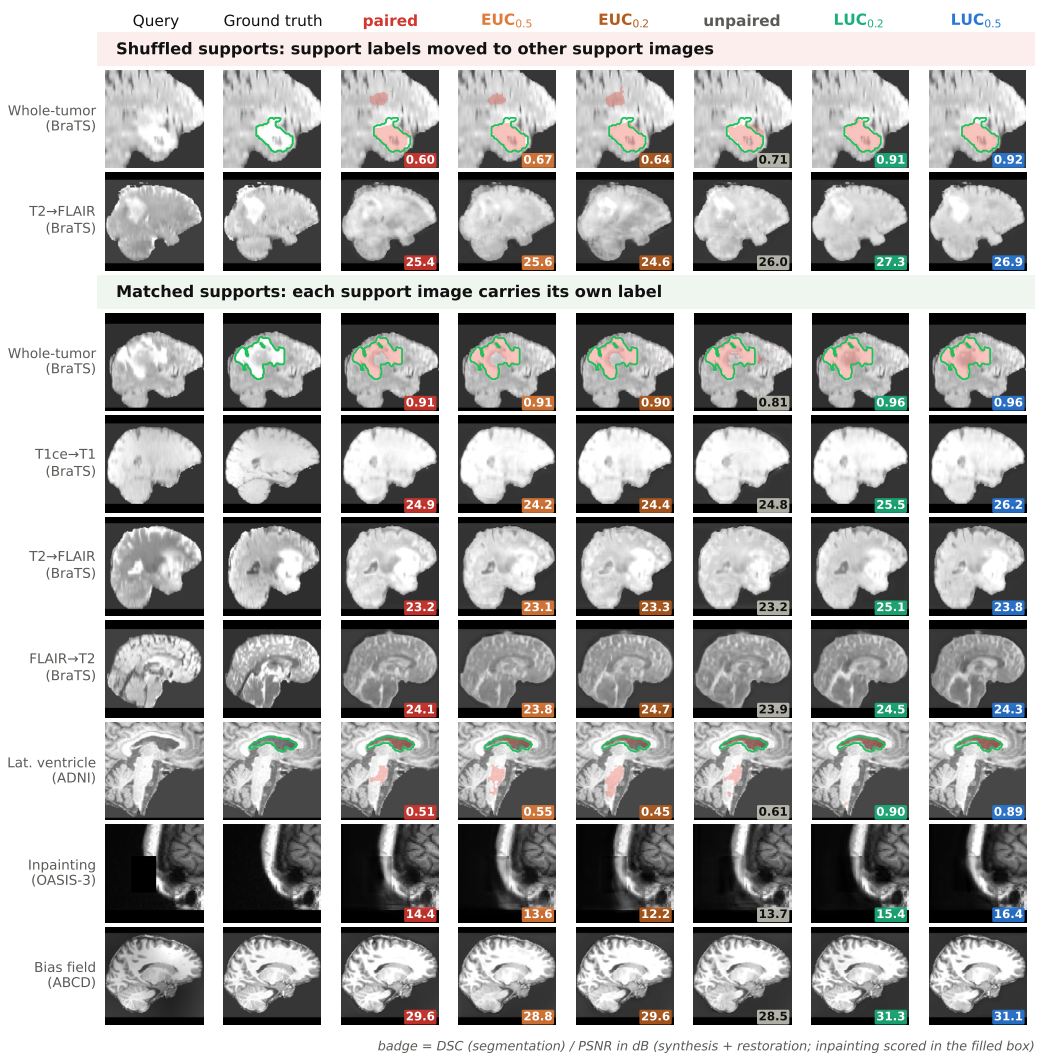


Figure 9: **Qualitative comparison across tasks and models** ($K$=4)**.** Six models (paired → $\text{EUC}_{0.5}$ → $\text{EUC}_{0.2}$ → unpaired → $\text{LUC}_{0.2}$ → $\text{LUC}_{0.5}$); green = ground-truth contour, red = predicted mask; the corner badge is DSC (segmentation) or PSNR in dB (synthesis/restoration; inpainting scored in the filled box only). Rows are labeled task (dataset). **Top, shuffled supports:** two tasks; the paired model degrades (partial or displaced mask, lost detail) while $\text{LUC}_{0.5}$ and $\text{LUC}_{0.2}$ hold, $\text{EUC}_{0.5}$ and $\text{EUC}_{0.2}$ degrade with the paired model, and the unpaired model sits in between. **Bottom, matched supports:** the trained task suite across all four datasets. Accuracy is comparable throughout: $\text{LUC}_{0.5}$ matches the paired model on translation and restoration, is cleaner on the two segmentation tasks (whole-tumor over-segmentation; atrophic ADNI ventricle), and fills the inpainting box slightly better. The bottom-row cases were selected to show the pattern; population numbers are in Tables 6 and 7.

A mis-registration of the support labels by 8 voxels along each of two axes costs the paired model 0.035 DSC and $\text{LUC}_{0.5}$ 0.001.

## E SCHEDULE ORDER

**Comparison design.** The two time-reversed pairs of Table 4 unpair for the same number of epochs and differ only in temporal order, and every other training setting is fixed across the family, so the two runs of a pair differ only in the timing of the unpairing.

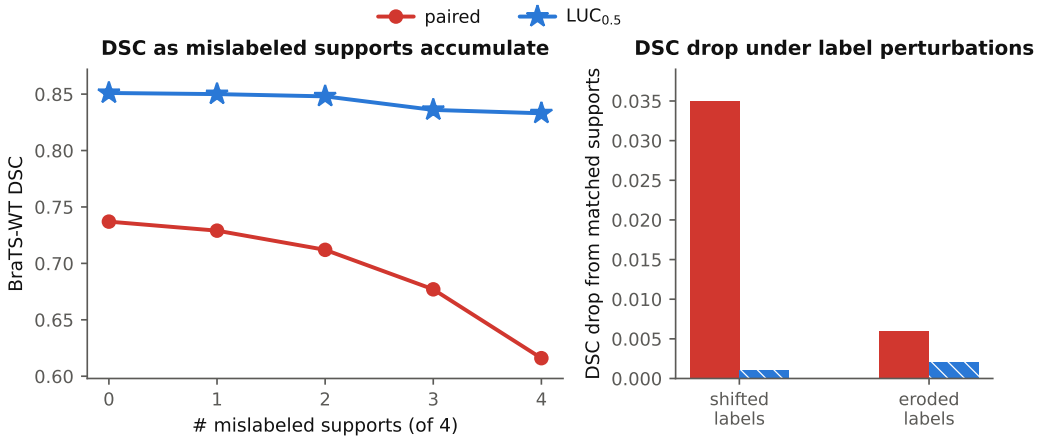


Figure 10: **Score under replaced, mis-registered, and eroded support labels.** Left: BraTS whole-tumor DSC as 0 to 4 of the $K{=}4$ support labels are replaced by another patient's label. Right: DSC drop from the matched-support score under a support mis-registration and a partial erosion of the support masks. Neuroverse3D, $n{=}60$.

**The six models.** Figure 4 gives matched-support DSC and the pairing gap of all six models on BraTS whole tumor at $K{=}4$. With the same $x$, the gap follows whether training ends on matched or on unpaired supports, at $x{=}0.5$ and at $x{=}0.2$. $\text{EUC}_{0.2}$, whose 30 unpairing epochs are followed by 120 matched epochs, ends with a gap of $-0.130$: the gap comes back once training returns to matched supports.

**Within-run trajectory.** The per-epoch validation gap shows the timing directly (Figure 6; per-task trajectories of $\text{LUC}_{0.2}$ in Figure 12). $\text{EUC}_{0.5}$ has a gap of about 0 through epoch 74, and the gap opens right after its switch to matched supports at epoch 75. $\text{LUC}_{0.5}$ carries the whole-tumor gap of the paired model before its switch ($-0.153$ over epochs 60 to 74); after the switch to unpaired supports at epoch 75 the gap is within 0.02 of zero in five epochs and averages $-0.007$ over epochs 90 to 149, closing about as fast as the gap of $\text{EUC}_{0.5}$ opens. $\text{EUC}_{0.2}$ has a gap of about 0 during its unpaired phase, and the gap opens once training returns to matched supports. The gap of $\text{LUC}_{0.2}$ shrinks from $-0.124$ to $-0.013$ within one epoch of its switch at epoch 120. On each of the five tasks tracked during training, the gap of $\text{LUC}_{0.5}$ shrinks by 71 to 97% after the switch (Figure 6b), and Figure 12 shows the same for $\text{LUC}_{0.2}$. Pairing dependence tracks the current training distribution: it follows the most recent phase and rises and falls at similar rates.

**LUC and EUC on three probes.** Figure 11 compares the two mirror pairs on the injected-mask response, the DSC lost under the derangement (Table 9) and the support sensitivity, the per-voxel standard deviation of the prediction inside the true tumor across four independently drawn support sets. For $\text{LUC}_{0.5}$ against $\text{EUC}_{0.5}$ these are 0.095 against 0.817, 0.012 against 0.088 and 0.046 against 0.074, and for $\text{LUC}_{0.2}$ against $\text{EUC}_{0.2}$ they are 0.340 against 0.801, 0.001 against 0.101 and 0.034 against 0.071.

## F Probes of how the models use the support set

The probes in this section measure how the predictions respond to perturbed support labels. Both compare all six models, and each states its sample size.

### Injected-mask response

The injected-mask test adds the sphere of Section A to every support label and measures the change in the fraction of sphere voxels predicted as foreground ($n{=}25$; Figure 8). The values order by what training ends on: paired $+0.846$, $\text{EUC}_{0.5}$ and $\text{EUC}_{0.2}$ (end on matched supports) $+0.817$ and

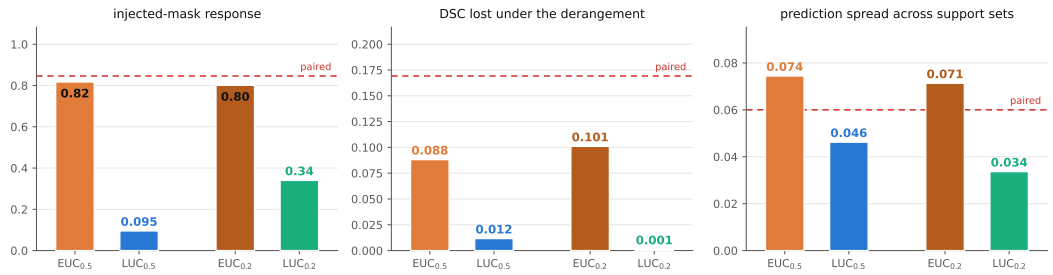


Figure 11: **LUC versus EUC at $x$=0.5 and $x$=0.2.** Injected-mask response, DSC lost under the derangement and support sensitivity; the two models of a pair unpair for the same number of epochs, the LUC model ends on unpaired supports and its EUC mirror on matched supports, and the dashed line is the paired model.

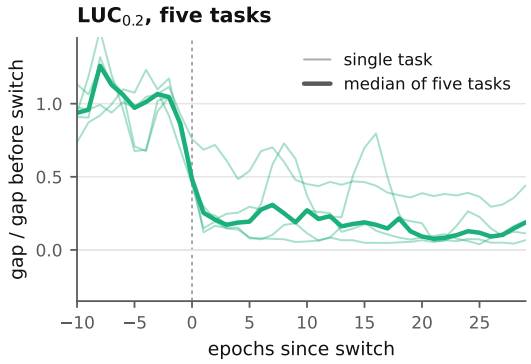


Figure 12: **Pairing gap of $\text{LUC}_{0.2}$ after its switch at epoch 120.** The gap on each of five tasks, divided by its mean over the ten epochs before the switch; thin lines for tasks and the thick line for their median.

$+0.801$, $\text{LUC}_{0.2}$ $+0.340$, and $\text{LUC}_{0.5}$ (ends on unpaired supports) $+0.095$. The unpaired model responds with $+0.687$: unpairing throughout gives it a positive gap, yet it reproduces most of the injected mask, while $\text{LUC}_{0.5}$ brings both close to zero. The ordering puts the models that end on matched supports (paired and both EUC models) at the high end and the two LUC models at the low end.

### THREE SUPPORT-LABEL CONDITIONS

Table 9 evaluates BraTS whole-tumor DSC ($n$=40) under three support-label conditions. `correct` leaves the supports as they are. `deranged` moves the labels to other support images, with the multiset unchanged. `rolled16` shifts each label 16 voxels relative to its image. A model that depends on the assignment scores highest at `correct` and lower off it; a model that does not is flat across the three. The paired model drops from 0.754 to 0.585 at `deranged`, and both EUC models behave the same way ($0.755 \to 0.667$ and $0.751 \to 0.650$). Both LUC models are flat and high across all three conditions ($\geq$0.848). The unpaired model is lowest under correct labels and rises under derangement, because a matched support set is itself out-of-distribution for a model trained only on unpaired supports.

The probes agree in direction. The two LUC models, which end on unpaired supports after a matched phase, hold their DSC across the three support-label conditions and give the weakest injected-mask response. The paired model and both EUC models, which end on matched supports, lose DSC when the labels move and reproduce the injected mask. The two models of each mirror pair unpair for the same number of epochs and differ in when the unpairing is applied, so what separates them is what training ends on. $\text{LUC}_{0.5}$ combines a small pairing gap and a weak injected-mask response with matched-support DSC 0.857 against 0.733 for the paired model.

Table 9: **Three support-label conditions** (BraTS whole-tumor DSC, $n$=40). The paired model and both EUC models lose DSC at `deranged`; $\text{LUC}_{0.5}$ and $\text{LUC}_{0.2}$ are flat across all three support-label conditions.

| model | correct | deranged | rolled16 |
|---|---|---|---|
| paired | 0.754 | 0.585 | 0.685 |
| unpaired | 0.630 | 0.741 | 0.710 |
| $\text{EUC}_{0.5}$ | 0.755 | 0.667 | 0.686 |
| $\text{EUC}_{0.2}$ | 0.751 | 0.650 | 0.699 |
| $\text{LUC}_{0.5}$ | **0.860** | **0.848** | 0.852 |
| $\text{LUC}_{0.2}$ | **0.880** | **0.879** | 0.880 |

## G TASKS OUTSIDE TRAINING

We evaluate the paired model and both LUC models on tasks that never occur in training, with the protocol of Section D: $K$=4 supports fixed per dataset from its training split, queries from its validation split (40 for BraTS and IXI[1], 20 per T1 cohort), matched and shuffled supports, DSC for segmentation, and PSNR inside the target foreground or, for inpainting, inside the filled boxes. One derangement is drawn per query and shared by all models. Held-out tasks change the segmentation target, the translation direction, the inpainting setting or the bias-field grid, or apply inpainting to BraTS and IXI; the held-out segmentation targets span both hemispheres or several structures, while training used single structures. Cross-dataset episodes take the supports from one dataset and the queries from another, which never happens in training. Table 10 lists 53 tasks, grouped by task type: the tasks on which all three models perform the task, with DSC of at least 0.6 or at least 1 dB above the unprocessed input, and at least one LUC model scores significantly above the paired model while neither scores significantly below it, together with three bias-correction tasks with a coarser $3^3$ control grid (on the T1 cohorts, BraTS→ABCD and ABCD→BraTS) on which the paired model stays within 0.2 dB of the input while both LUC models improve on it by 0.7 to 1.3 dB. On 46 of the 53 tasks both LUC models score significantly above the paired model (subject-level Wilcoxon, $p$<0.05); on the seven marked ones, one of them does and the other does not differ significantly from it. On the 46 tasks, $\text{LUC}_{0.5}$ scores up to 0.128 DSC and 3.8 dB above the paired model, and its pairing gap averages −0.004 DSC and −0.13 dB, against −0.054 DSC and −2.41 dB for the paired model.

[1] `https://brain-development.org/ixi-dataset/`

Table 10: **Tasks outside training.** Matched-support score (DSC; PSNR in dB, inside the filled boxes for inpainting) and pairing gap; "input" is the PSNR of the unprocessed input, and an arrow in the data column marks a cross-dataset episode. Both LUC models score significantly above the paired model on every unmarked task (subject-level Wilcoxon, $p<0.05$); on tasks marked $^\circ$, one LUC model does and the other does not differ significantly from it. $^\dagger$Shown in Figure 5.

| task | data | $n$ | input | paired ($p$=0) | | $\text{LUC}_{0.5}$ | | $\text{LUC}_{0.2}$ | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | matched | gap | matched | gap | matched | gap |
| *Segmentation (DSC)* | | | | | | | | | |
| Hippocampus$^\dagger$ | T1 cohorts | 60 | – | 0.608 | −0.011 | 0.706 | +0.004 | 0.707 | −0.017 |
| Caudate | T1 cohorts | 60 | – | 0.632 | −0.014 | 0.720 | −0.004 | 0.716 | −0.014 |
| Thalamus$^\dagger$ | T1 cohorts | 60 | – | 0.758 | −0.001 | 0.853 | 0.000 | 0.832 | −0.002 |
| Edema from FLAIR$^\dagger$ | BraTS | 40 | – | 0.627 | −0.088 | 0.695 | −0.003 | 0.711 | −0.004 |
| Putamen$^\dagger$ | T1 cohorts | 60 | – | 0.672 | −0.019 | 0.746 | −0.007 | 0.748 | −0.013 |
| Lateral ventricles$^\dagger$ | T1 cohorts | 60 | – | 0.817 | −0.034 | 0.893 | −0.003 | 0.892 | −0.004 |
| Ventricular system | T1 cohorts | 60 | – | 0.792 | −0.042 | 0.856 | −0.005 | 0.854 | −0.009 |
| Cerebral white matter | T1 cohorts | 60 | – | 0.849 | −0.115 | 0.890 | 0.000 | 0.889 | −0.001 |
| Cerebellum and brainstem | T1 cohorts | 60 | – | 0.888 | −0.008 | 0.904 | 0.000 | 0.913 | −0.007 |
| Cerebellum | T1 cohorts | 60 | – | 0.872 | −0.007 | 0.888 | −0.002 | 0.896 | −0.007 |
| Basal ganglia, left$^\circ$ | T1 cohorts | 60 | – | 0.699 | −0.009 | 0.762 | −0.006 | 0.737 | −0.011 |
| Lateral ventricle, left$^\dagger$ | ABCD→ADNI | 20 | – | 0.693 | −0.164 | 0.821 | −0.037 | 0.759 | −0.071 |
| White matter, left$^\dagger$ | ADNI→ABCD | 20 | – | 0.820 | −0.156 | 0.894 | 0.000 | 0.869 | −0.002 |
| Thalamus, left$^\dagger$ | ABCD→ADNI | 20 | – | 0.748 | −0.004 | 0.819 | 0.000 | 0.777 | −0.002 |
| White matter, left | ABCD→ADNI | 20 | – | 0.849 | −0.206 | 0.886 | −0.003 | 0.891 | −0.001 |
| Brain mask$^\dagger$ | ABCD→OASIS-3 | 20 | – | 0.677 | −0.040 | 0.699 | −0.008 | 0.732 | −0.012 |
| Brain mask | OASIS-3→ABCD | 20 | – | 0.617 | 0.000 | 0.622 | 0.000 | 0.662 | 0.000 |
| Brain mask$^\dagger$ | ADNI→OASIS-3 | 20 | – | 0.723 | −0.010 | 0.744 | +0.005 | 0.741 | 0.000 |
| *Modality translation (PSNR)* | | | | | | | | | |
| T2→T1ce$^\dagger$ | BraTS | 40 | 16.04 | 19.80 | −0.26 | 20.33 | −0.04 | 20.50 | −0.10 |
| T1ce→T2$^\dagger$ | BraTS | 40 | 16.22 | 20.42 | −0.14 | 20.73 | +0.05 | 20.74 | +0.05 |
| FLAIR→T1$^\circ$ | BraTS | 40 | 19.55 | 21.60 | −1.43 | 21.73 | −0.09 | 22.05 | −0.11 |
| T2→T1$^\dagger$ | IXI→BraTS | 40 | 14.90 | 16.08 | −0.34 | 17.66 | −0.12 | 16.89 | −0.14 |
| *Inpainting (PSNR inside the filled boxes)* | | | | | | | | | |
| Inpainting, T1$^\dagger$ | BraTS | 40 | 3.87 | 16.42 | −3.13 | 18.80 | −0.18 | 18.33 | −0.45 |
| Inpainting, T1ce | BraTS | 40 | 4.43 | 16.09 | −3.38 | 17.98 | −0.34 | 16.81 | −0.61 |
| Inpainting, FLAIR$^\dagger$ | BraTS | 40 | 4.59 | 16.42 | −3.78 | 17.68 | −0.15 | 17.44 | −0.55 |
| Inpainting, FLAIR, 5–6 boxes of side 12–20 | BraTS | 40 | 5.19 | 17.71 | −3.58 | 18.84 | −0.39 | 18.62 | −0.77 |
| Inpainting, T1ce, one box of side 56–64 | BraTS | 40 | 3.55 | 14.95 | −2.49 | 16.66 | −0.14 | 15.12 | −0.29 |
| Inpainting, T1ce, 5–6 boxes of side 12–20 | BraTS | 40 | 5.06 | 17.90 | −3.62 | 18.91 | −0.58 | 18.43 | −0.88 |
| Inpainting, FLAIR, one box of side 56–64 | BraTS | 40 | 3.99 | 15.04 | −3.02 | 16.06 | −0.08 | 15.46 | −0.27 |
| Inpainting, T2$^\dagger$ | IXI | 40 | 9.81 | 15.93 | −0.70 | 16.38 | −0.05 | 16.66 | −0.03 |
| Inpainting, 5–6 boxes of side 12–20 | T1 cohorts | 60 | 6.50 | 18.84 | −2.13 | 19.34 | +0.03 | 19.32 | −0.02 |
| Inpainting, one box of side 56–64 | T1 cohorts | 60 | 4.71 | 16.16 | −1.34 | 16.45 | +0.01 | 16.40 | −0.02 |
| Inpainting, T2$^\circ$ | BraTS | 40 | 6.09 | 17.25 | −3.83 | 18.00 | −0.20 | 17.58 | −0.41 |
| Inpainting, T1$^\circ$ | IXI | 40 | 8.41 | 16.35 | −1.44 | 16.23 | +0.10 | 16.70 | −0.10 |
| Inpainting$^\dagger$ | IXI→ABCD | 20 | 4.69 | 11.51 | −1.59 | 15.31 | −0.05 | 14.95 | −0.40 |
| Inpainting | OASIS-3→ABCD | 20 | 4.69 | 11.11 | −0.08 | 14.13 | +0.01 | 13.37 | −0.01 |
| Inpainting$^\dagger$ | ABCD→IXI | 40 | 8.41 | 9.61 | +1.63 | 12.10 | +0.24 | 12.06 | +0.08 |
| Inpainting | OASIS-3→ADNI | 20 | 4.76 | 11.85 | −0.08 | 14.16 | 0.00 | 13.57 | 0.00 |
| Inpainting | ADNI→IXI | 40 | 8.41 | 12.13 | +1.09 | 13.47 | +0.33 | 14.11 | +0.18 |
| Inpainting$^\dagger$ | BraTS→ABCD | 20 | 4.69 | 16.78 | −2.59 | 18.40 | −0.08 | 18.11 | −0.24 |
| Inpainting | ADNI→ABCD | 20 | 4.69 | 16.09 | −4.26 | 17.45 | +0.09 | 17.15 | +0.04 |
| Inpainting$^\dagger$ | ABCD→BraTS | 40 | 3.87 | 17.78 | −3.53 | 18.86 | −0.04 | 18.74 | −0.12 |
| Inpainting | OASIS-3→IXI | 40 | 8.41 | 12.47 | +0.14 | 13.42 | +0.03 | 13.05 | +0.04 |
| Inpainting$^\circ$ | BraTS→ADNI | 20 | 4.76 | 15.95 | −2.27 | 16.22 | +0.03 | 16.86 | −0.13 |
| Inpainting$^\circ$ | ABCD→ADNI | 20 | 4.76 | 16.54 | −2.45 | 16.58 | +0.13 | 17.29 | −0.06 |
| *Bias correction (PSNR)* | | | | | | | | | |
| Bias correction, $3^3$ grid, log-strength 0.4–0.8$^\dagger$ | T1 cohorts | 60 | 25.67 | 26.92 | −4.32 | 27.99 | −0.18 | 27.75 | −0.04 |
| Bias correction, $3^3$ grid, log-strength 0.5–0.9 | T1 cohorts | 60 | 24.24 | 26.17 | −3.84 | 27.50 | −0.18 | 26.74 | −0.03 |
| Bias correction, $3^3$ control grid$^\dagger$ | T1 cohorts | 60 | 27.67 | 27.57 | −4.84 | 28.42 | −0.21 | 28.41 | −0.08 |
| Bias correction, $3\times4\times4$ grid | T1 cohorts | 60 | 25.84 | 27.04 | −4.25 | 27.99 | −0.16 | 27.22 | −0.09 |
| Bias correction, $3^3$ control grid$^\dagger$ | BraTS→ABCD | 20 | 25.86 | 26.05 | −6.58 | 26.83 | −1.31 | 27.19 | −2.17 |
| Bias correction, $3^3$ control grid$^\dagger$ | ADNI→ABCD | 20 | 25.86 | 26.98 | −6.45 | 28.17 | −0.20 | 27.72 | −0.03 |
| Bias correction, $3^3$ control grid$^\dagger$ | ABCD→BraTS | 40 | 25.52 | 25.57 | −2.49 | 26.39 | +0.05 | 26.30 | −0.30 |
| Bias correction$^\circ$ | ABCD→ADNI | 20 | 24.57 | 25.69 | −5.67 | 26.17 | −0.35 | 26.29 | −0.90 |